SRINIVAS UNIVERSITY

INSTITUTE OF ENGINEERING & TECHNOLOGY

IOT BASED MINI PROJECT

SMART STREET LIGHT AUTOMATION SYSTEM FOR MINIMIZING POWER CONSUMPTION

Submitted By:-

Akarsha P 1SU21CS008

Bipin T P 1SU21CS021

Gagan N Salian 1SU21CS036

Nikhil C N 1SU21CS081

Submitted To:-

Janapathi Venkata Krishna

Department Of CSE

Contents

Abstract

In recent years the demand is increasing day to day. It is found that a large quantity of electricity in many countries is consumed by lighting the streets. In this project, the system is to develop a street light energy-saving control system to reduce energy if no vehicles pass through certain roads. The operation of this system is to maintain the intensity of street lighting to 40% of the maximum intensity if no vehicles pass through the road. When the IR sensor detects movement of the vehicle, the street lights will be switched to 100% intensity. LDR is used in the system to detect day/night.

In this project, Arduino Microcontroller is used to build the smart street light system. Once the day is detected all the streetlights will be in the OFF state. Arduino microcontroller is used to control the system. IR sensor functions as a vehicle detector that will send a signal to the Arduino will control the intensity of the LED while the current sensor is used as the current detector LED lamp.

The prototype for the street lighting energy-saving control system also has safety usage in that the light will not turn OFF completely but only dim and the user can easily see from far away and the light will fully turn ON if it detects movement. The system has shown great energy savings and if the system can be upgraded with many functions and user friendly the system can be commercialized and the cost of retrofitting the street lighting energy-saving control system can be lowered. This system also focuses on time dependency, short-term atmospheric changes, traffic light compatibility, and flexibility of the smart street light system.

Introduction

Smart street light refers to a network-connected street light. Each street light is equipped with an outdoor lamp controller, an internet of Things device, and or sensors. A smart street light automatically regulates the light intensity based on sunset or sunrise times, daily schedule, human presence and weather situation Thereby by saving considerable energy and lowering maintenance costs.The system is designed in such a way that the street lights are in off state during the day on-state at night based on LDR. At night the street lights are at minimum intensity. If any object or vehicle is detected by the IR sensor the street light glows at maximum intensity, further if not detected light remains in minimum intensity.

Benefits of Smart Street Lighting are Smart street lights are a cost-effective, practical, and sustainable way to minimize energy usage and expenses.Smart street lights also improve the quality of life and citizen safety in the community by alerting the lighting operator in time for a prompt repair where necessary. In an existing system using normal street lamps, takes more current and is expensive so we must use LED lamps to save the current, and a low amount of power is required. Using IoT-type systems is all over the world. It used to be watched in all kinds of areas in the cities.

The objective of the project is to save power consumption by street light and design and independent automated system to detect an object by using sensors based on time and weather conditions, also to develop an iot based autonomous system to monitor and set the intensity of street light using blink app. The system has shown great energy savings and if the system can be upgraded with many functions and userfriendly concepts like humidity and temperature sensors,traffic light compatible, autoerror detection and piezoelectric effect.The system can be commercialized and the cost for retrofitting the street lighting energy-saving control system can be lowered.

System Requirements:-

Hardware Requirements:-

Processor: 10th generation Intel core i3 processor

RAM: 8GB

Hybrid Storage:1TB (512GB HDD+512GB SSD)

Software Requirements:-

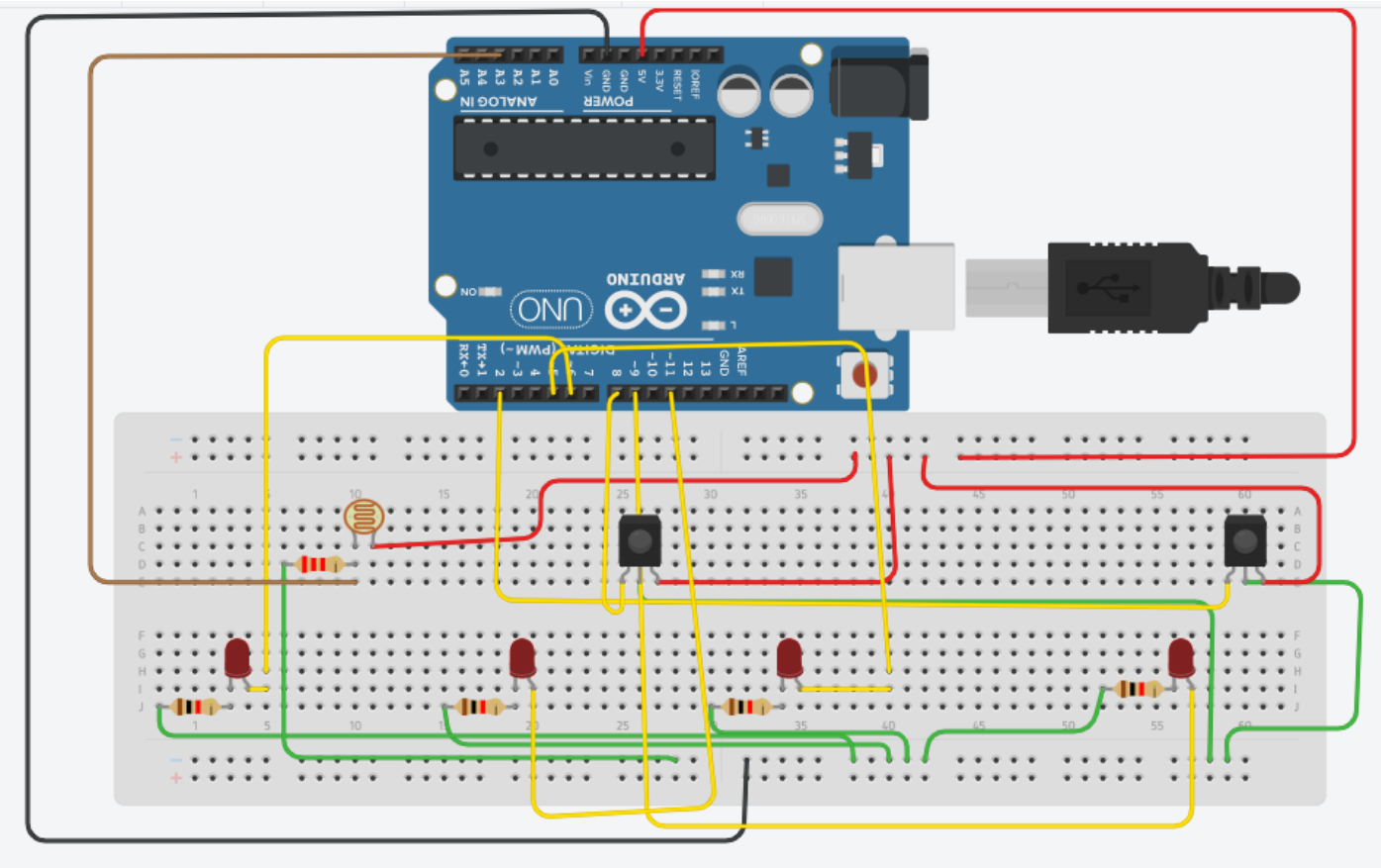
Operating System: Windows 10 Pro

Application: Arduino 2.0.4

Components Required:-

- Arduino Uno
- IR Sensors
- LED
- LDR
- Resistors used 3K3,1K
- Bread Board
- Jumper Wires

Design & Architecture



Implementation

Phase 1:-

```
int led=3;
int led2=5;
int ir1=2;
int ir2=6;
int ldr=A3;
int ts=40;
void setup() {
  // put your setup code here, to run once:
  pinMode(led,OUTPUT);
  pinMode(led2,OUTPUT);
  pinMode(ir1,INPUT);
  pinMode(ir2,INPUT);
  pinMode(ldr,INPUT);
  Serial.begin(9600);
}
void loop() {
  // put your main code here, to run repeatedly:
int d=analogRead(ldr);
int i=digitalRead(ir1);
int j=digitalRead(ir2);
    if(d<100){
    if(i==LOW ){
      digitalWrite(led,HIGH);
      delay(1000);
```

```
  }
  if(j==LOW ){
    digitalWrite(led2,HIGH);
    delay(1000);
  }
  else{
    digitalWrite(led,LOW);
    digitalWrite(led2,LOW);
  }
  }
  else{
    digitalWrite(led,LOW);
    digitalWrite(led2,LOW);
  }
}
```

Phase 2:-

```
#include <Time.h>
#include <TimeLib.h>
int led=3;
int led2=5;
int ir1=2;
int ir2=6;
int ldr=A3;
int ts=40;
int s;
int m;
```

```
int d=1,mo=2,y=2002;
void setup()
{
  Serial.begin(9600);
  setTime(0,0,0,d,mo,y);
  pinMode(led,OUTPUT);
  pinMode(led2,OUTPUT);
  pinMode(ir1,INPUT);
  pinMode(ir2,INPUT);
  pinMode(ldr,INPUT);
}
void loop()
{
  digitalClockDisplay();
  delay(1000);
   m=minute();
   s=second();
   if((m==3)&&(s==59))
   {
    setTime(0,0,0,d,mo,y);
    d++;
   }
  int d=analogRead(ldr);
  int i=digitalRead(ir1);
  int j=digitalRead(ir2);
  Serial.print(" ");
  Serial.print(d);
```

```
  if((d<100)&&(m>=1)){
    analogWrite(led,25);
    analogWrite(led2,25);
    if(i==LOW ){
      digitalWrite(led,HIGH);
    }
    if(j==LOW ){
      digitalWrite(led,HIGH);
       digitalWrite(led2,HIGH);
    }
  }
    else{
      digitalWrite(led,LOW);
      digitalWrite(led2,LOW);
      Serial.print(" ");
      Serial.print("off");
    }
}
void digitalClockDisplay()
{
  printDigits(m);
  printDigits(s);
  Serial.print(" ");
  Serial.print(day());
  Serial.print(" ");
  Serial.print(month());
  Serial.print(" ");
```

```
  Serial.print(year());
  Serial.println();
}
void printDigits(int digits)
{
  Serial.print(":");
  if(digits < 10)
    Serial.print('0');
  Serial.print(digits);
}
```

Phase 3:-

```
#define BLYNK_PRINT Serial
#define BLYNK_TEMPLATE_ID "TMPL33UAiZaTI"
#define BLYNK_TEMPLATE_NAME "iot"
#define BLYNK_AUTH_TOKEN "31QDzKMsGtznN9P5hozAzByTi97VfEw7"
#define BLYNK_PRINT DebugSerial
#include <SoftwareSerial.h>
SoftwareSerial DebugSerial(2, 3); // RX, TX
#include <Time.h>
#include <TimeLib.h>
#include <BlynkSimpleStream.h>
char auth[] = "31QDzKMsGtznN9P5hozAzByTi97VfEw7";
BlynkTimer timer;
int comin=0;
int comax=0;
```

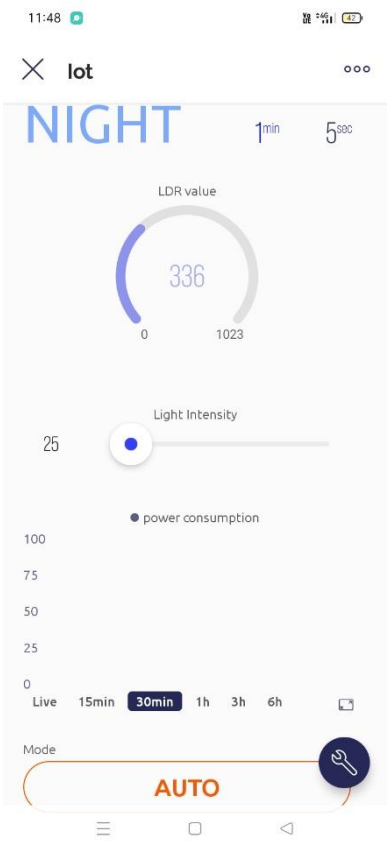
```
int iten=25;
int led=9;
int led2=5;
int led3=11;
int led4=6;
int ir1=2;
int ir2=8;
int ldr=A3;
int s;
int m;
int d=1,mo=2,y=2002;
int man;
void setup()
{
  DebugSerial.begin(9600);
  setTime(0,0,0,d,mo,y);
  pinMode(led,OUTPUT);
  pinMode(led2,OUTPUT);
  pinMode(led3,OUTPUT);
  pinMode(led4,OUTPUT);
  pinMode(ir1,INPUT);
  pinMode(ir2,INPUT);
  pinMode(ldr,INPUT);
  digitalWrite(led,LOW);
  digitalWrite(led2,LOW);
  digitalWrite(led3,LOW);
  digitalWrite(led4,LOW);
```

```
  Serial.begin(9600);
  Blynk.begin(Serial, auth);
  timer.setInterval(0L,smrt);
}
BLYNK_WRITE(V0){
  iten=param.asInt();
}
BLYNK_WRITE(V2){
  man=param.asInt();
}
   void smrt(){
   m=minute();
   s=second();
   char st[5]="DAY";
   while((m==3)&&(s==59))
   {
    setTime(0,0,0,d,mo,y);
    int est=85*comin;
    int us=((25*comin)+(255*comax))/3;
    int cons=(us*100)/est;
    Blynk.virtualWrite(V1, cons);
      }
   if(m<1){
     Blynk.virtualWrite(V3,"DAY");
   }
   else{
     Blynk.virtualWrite(V3,"NIGHT");
```

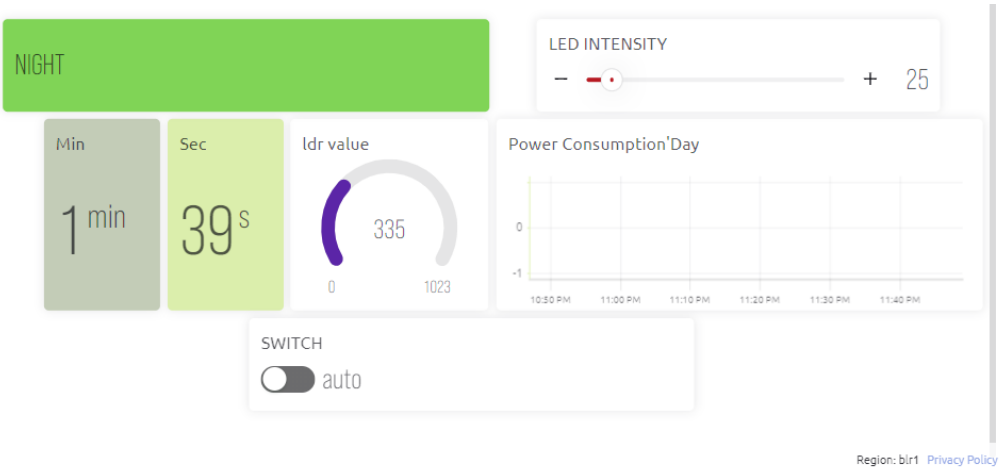
```
 }
 Blynk.virtualWrite(V7, m);
 Blynk.virtualWrite(V8, s);
int v=analogRead(ldr);
Blynk.virtualWrite(V6, v);
int i=digitalRead(ir1);
int j=digitalRead(ir2);
if((v<100)&&(m>=1)&&(man==0))
{
  comin++;
  analogWrite(led,iten);
  analogWrite(led2,iten);
  analogWrite(led3,iten);
  analogWrite(led4,iten);
  if(i==LOW ){
    digitalWrite(led,HIGH);
    digitalWrite(led2,HIGH);
    delay(500);
    comax++;
  }
  if(j==LOW ){
     digitalWrite(led3,HIGH);
     digitalWrite(led4,HIGH);
     delay(500);
     comax++;
  }
}
```

```
   else{
      digitalWrite(led,LOW);
      digitalWrite(led2,LOW);
      digitalWrite(led3,LOW);
      digitalWrite(led4,LOW);
    }
  if((m<1)&&(v<100))
    {
  Blynk.logEvent("system_alert") ;
    }
  if(man==1)
   {
      analogWrite(led,iten);
      analogWrite(led2,iten);
      analogWrite(led3,iten);
      analogWrite(led4,iten);
   }
   }
void loop()
{
  Blynk.run();
  timer.run();
}
```

Evaluation & Results



Blynk App Project Interface



Blynk Website Project Web Interface

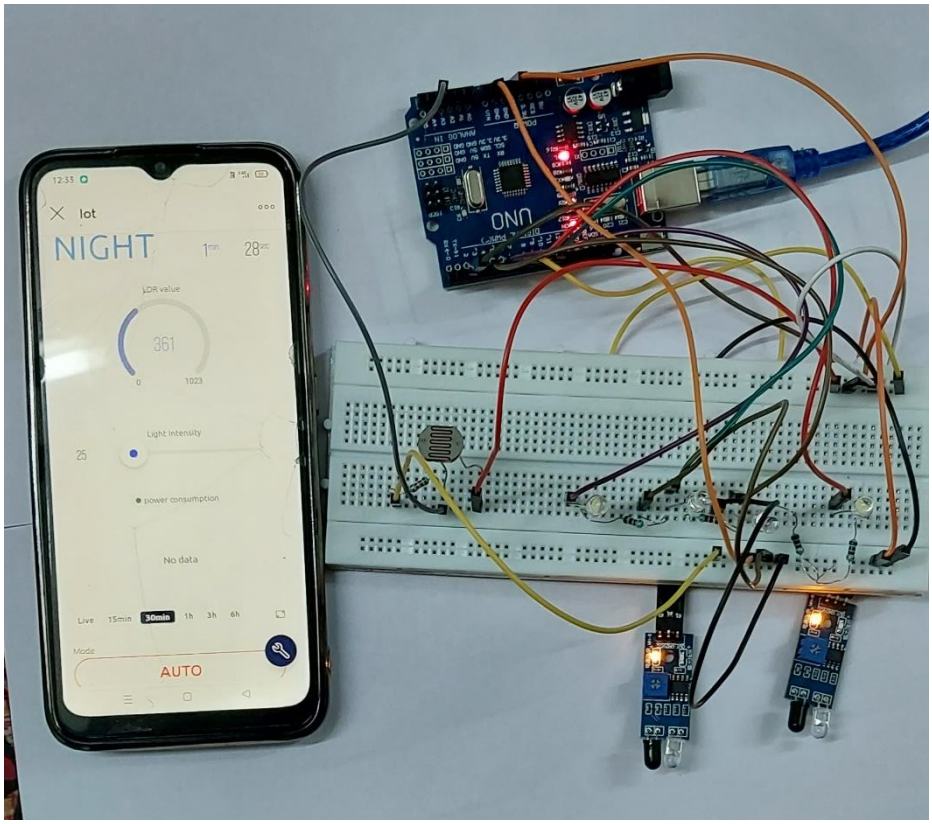


Fig 1:Light off due to high visibility

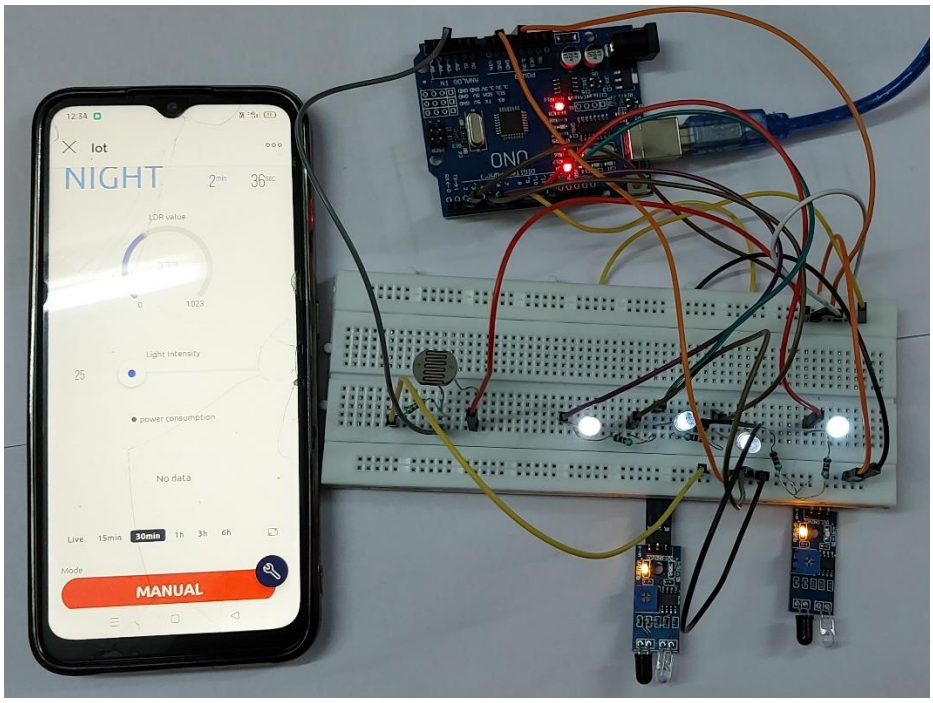


Fig 2:Manually switching all lights

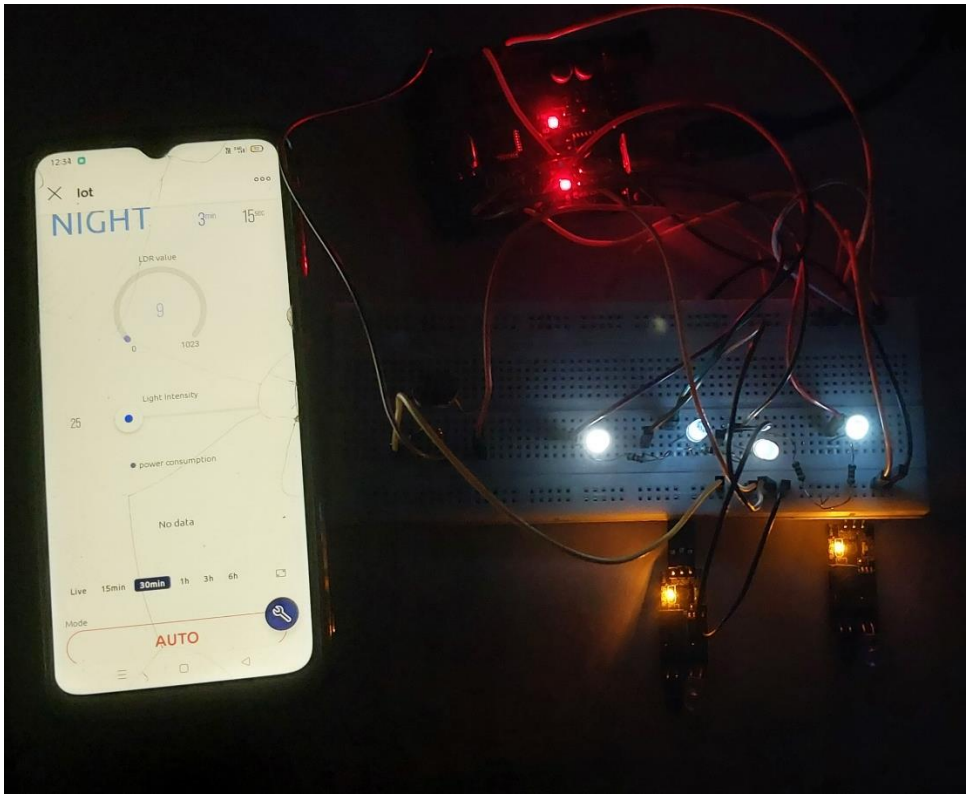


Fig 3:Lights on low visibility with minimum intensity

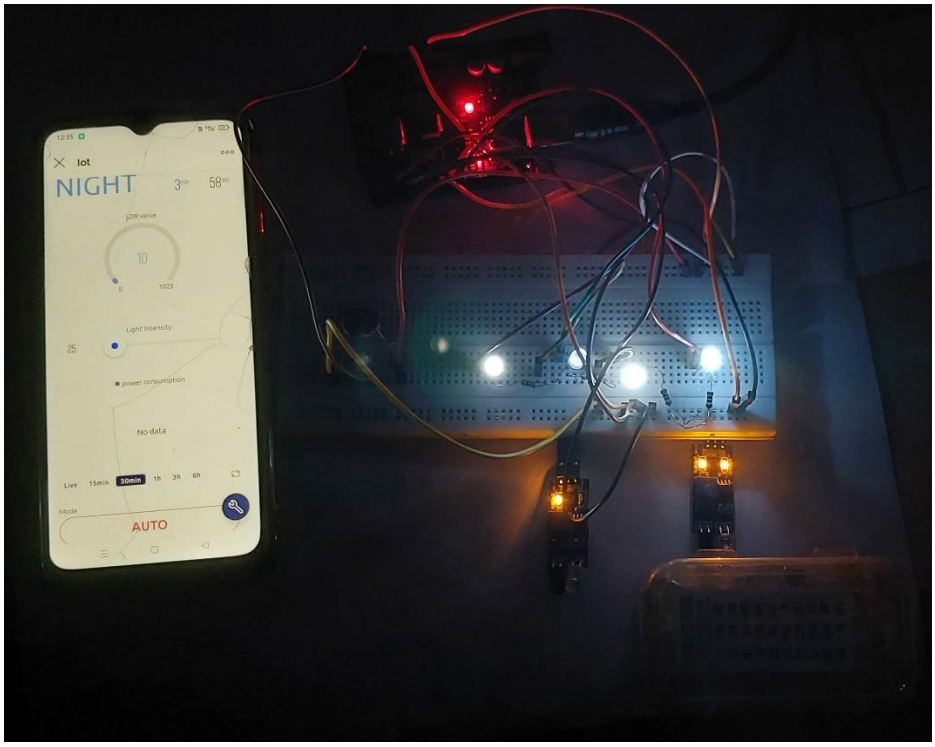


Fig 4:The LED glows with maximum intensity on object detection